

PRÍKLADY NA PRECVIČENIE SYNCHRONIZAČNÉHO PROBLÉMU
PRODUCENT – KONZUMENT V JAZYKU C/C++ S VYUŽITÍM
VLÁKIEN, MUTEXOV A PODMIENKOVÝCH PREMENNÝCH

Obsah

Zadanie 1 – Prvočíselný rozklad	3
Zadanie 2 – Kombinačné čísla	4
Zadanie 3 – Testovanie Fibonacciho čísel	5
Zadanie 4 – Kuchár – čašník	6
Zadanie 5 – Polícia – Sherlock Holmes	7
Zadanie 6 – Hercule Poirot – Traja podozriví	8
Zadanie 7 – Polícia – Columbo	9
Zadanie 8 – Teletubbies	10
Zadanie 9 – Pekáreň	11
Zadanie 10 – Bar	12
Zadanie 11 – Pátrač Otsok Shavk	13
Zadanie 12 – Dovož čaju pre Britské impérium	14
Zadanie 13 – Počítač s SM Minmonsom	15
Zadanie 14 – Lovci morských tvorov	16
Zadanie 15 – Zber čaju pre Britské impérium	17
Zadanie 16 – Hamutex Malloku	18
Zadanie 17 – BrandGames	19
Zadanie 18 – Plains of Shavk	20
Zadanie 19 – Voľby	21
Zadanie 20 – WaitChat	22
Zadanie 21 – Predvianočný zhon	23
Zadanie 22 – Káva pre doktoranda	24
Zadanie 23 – Výroba kávy	25
Zadanie 24 – Elektronické noviny	26
Zadanie 25 – Sušenie húb	27
Zadanie 26 – Sťahovacia spoločnosť	28
Zadanie 27 – Predaj hry	29
Zadanie 28 – 2Drinkstone	30
Zadanie 29 – Plavba na Oravskej priehrade	31
Zadanie 30 – Šialené odmeny	32

Zadanie 1 – Prvočíselný rozklad

V jazyku C/C++ vytvorte aplikáciu, ktorá bude mať jedno hlavné a jedno pomocné vlákno. Vlákna zdieľajú buffer s kapacitou 10 prvkov. Pomocné vlákno bude predstavovať producenta generujúceho n náhodných zložených čísel z intervalu $\langle a, b \rangle$. Tieto bude ukladať do buffra pracujúceho na princípe LIFO. Hlavné vlákno bude postupne vyberať jednotlivé čísla z buffra a pre každé z nich vypíše na obrazovku jeho prvočíselný rozklad. Aplikácia skončí, keď sa na obrazovku vypíše rozklad n zložených čísel. Parametre a , b , n zadáva používateľ ako argumenty programu. V prípade, že používateľ nezadá správny počet argumentov, na štandardný chybový výstup sa vypíše chybové hlásenie a aplikácia skončí.

Bodovanie:

- návrh potrebných dátových štruktúr: 3 body;
- správne vytvorenie a uvoľnenie synchronizačných primitív: 1 bod;
- producent: 5 bodov;
- konzument: 6 bodov;
- celková správnosť riešenia: 5 bodov.

V prípade, že sa Vaše zadanie **nebude dať skompilovať** na stroji frios2.fri.uniza.sk, automaticky dostávate **0 bodov**.

Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 2 – Kombinačné čísla

V jazyku C/C++ vytvorte aplikáciu, ktorá bude mať jedno hlavné a jedno pomocné vlákno. Vlákna zdieľajú buffer s kapacitou 20 prvkov. Pomocné vlákno bude predstavovať producenta generujúceho n dvojíc náhodných celých čísel z intervalu $\langle a, b \rangle$. Pomocné vlákno uloží vygenerovanú dvojicu čísel do buffra pracujúceho na princípe LIFO. Hlavné vlákno bude postupne vyberať jednotlivé dvojice z buffra a pre každú dvojicu (s, r) vypíše na obrazovku hodnotu kombinačného čísla, ktoré je možné získať z tejto dvojice. *Napr., ak máme dvojicu $(3, 7)$, väčšie číslo je 7, preto táto dvojica predstavuje kombinačné číslo $\binom{7}{3}$, ktorého hodnota je 35. V prípade dvojice $(5, 2)$ je väčšie číslo 5, preto táto dvojica predstavuje kombinačné číslo $\binom{5}{2}$, ktorého hodnota je 10.*

Aplikácia skončí, keď sa na obrazovku vypíše n kombinačných čísel. Parametre a, b, n zadáva používateľ ako argumenty programu. V prípade, že používateľ nezadá správny počet argumentov, na štandardný chybový výstup sa vypíše chybové hlásenie a aplikácia skončí.

Pri testovaní zadávajte argumenty a, b z množiny $\{1, 2, \dots, 12\}$, aby Vám nepretiekol dátový typ `int` pri výpočte faktoriálu.

Bodovanie:

- návrh potrebných dátových štruktúr: 3 body;
- správne vytvorenie a uvoľnenie synchronizačných primitív: 1 bod;
- producent: 5 bodov;
- konzument: 6 bodov;
- celková správnosť riešenia: 5 bodov.

V prípade, že sa Vaše zadanie **nebude dať skompilovať** na stroji `frios2.fri.uniza.sk`, automaticky dostávate **0 bodov**.

Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 3 – Testovanie Fibonacciho čísel

V jazyku C vytvorte aplikáciu, ktorá bude obsahovať jedno hlavné vlákno a dve pomocné vlákna. Hlavné vlákno spustí pomocné vlákna a bude čakať na ich ukončenie. Jedno vlákno bude generovať čísla od a po b a druhé vlákno bude kontrolovať, či je dané číslo z Fibonacciho postupnosti. Vlákno, ktoré generuje čísla, ich bude ukladať do buffra, ktorý môže obsahovať maximálne c čísel. Pokiaľ je buffer plný, vlákno musí počkať, pokiaľ sa buffer vyprázdni. Vlákno, ktoré kontroluje, či vygenerované číslo je Fibonacciho číslo, nemôže odoberať čísla z buffra pokiaľ je prázdny. Pokiaľ sa zistí, že číslo je Fibonacciho číslo, tak číslo sa vypíše.

Pomôcka:

- buffer môže byť reprezentovaný poľom v štruktúre.

Bodovanie:

- návrh potrebných dátových štruktúr: 3 body;
- správne vytvorenie a uvoľnenie synchronizačných primitív: 1 bod;
- producent: 5 bodov;
- konzument: 6 bodov;
- celková správnosť riešenia: 5 bodov.

V prípade, že sa Vaše zadanie **nebude dať skompilovať** na stroji frios2.fri.uniza.sk, automaticky dostávate **0 bodov**.

Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 4 – Kuchár – čašník

V jazyku C vytvorte aplikáciu, ktorá bude obsahovať jedno hlavné vlákno a dve pomocné vlákna. Hlavné vlákno predstavuje kuchára v reštaurácii. Pomocné vlákna predstavujú čašníkov. Kuchár pripravuje jedlá a odkladá ich na stolík, z ktorého ich čašníci berú a roznášajú. Na stolík sa zmestí maximálne zadaný počet jedál (ak počet nie je zadaný, tak na stolík sa zmestí 10 jedál). Roznos jednej porcie trvá približne od 2 do 5 sekúnd. Príprava jedla trvá od 0.5 do 1.5 sekundy. Čašník musí čakať v prípade, ak je stolík, z ktorého odoberá jedlá, prázdny. Kuchár čaká, ak je stolík plný.

Pomôcka:

- generovanie čísel zo zadaného intervalu:
`float r = rand() / (float) RAND_MAX; //<0,1>`
`r = min + r * (max - min); //<min, max>`

Bodovanie:

- návrh potrebných dátových štruktúr: 3 body;
- správne vytvorenie a uvoľnenie synchronizačných primitív: 1 bod;
- producent: 5 bodov;
- konzument: 6 bodov;
- celková správnosť riešenia: 5 bodov.

V prípade, že sa Vaše zadanie **nebude dať skompilovať** na stroji frios2.fri.uniza.sk, automaticky dostávate **0 bodov**.

Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 5 – Polícia – Sherlock Holmes

OCH NIE! Stal sa ohavný zločin, po ktorom bola polícia schopná zozbierať **25 kusov dôkazov** ukazujúcich na **2 rôznych páchatel'ov**. Preto si zavolali na pomoc známeho to detektíva Sherlocka Holmesa, ktorému postupne prezentujú dôkaz o vine jedného z podozrivých. Sherlock uzná za vinného toho z podozrivých, o ktorého vine svedčí viac dôkazov.



Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 6 – Hercule Poirot – Traja podozriví

OCH NIE! Stal sa ohavný čin typu vražda. Avšak Hercule Poirot bol nablízku a už vie, ktorý z troch podozrivých je vinníkom. Postupne hovorí o **dôkazoch** proti jednotlivým podozrivým. Ten podozrivý, proti ktorému svedčí najviac dôkazov, je vrahom!



Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

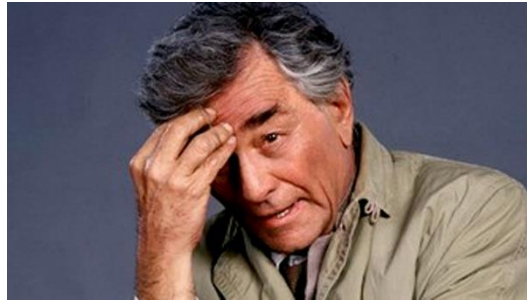
Zadanie 7 – Polícia – Columbo

OCH NIE! Stal sa ohavný zločin, po ktorom bola *polícia* schopná zozbierať **50 kusov** dôkazov ukazujúcich na **2 rôznych páchatel'ov** – niektoré z týchto dôkazov však neboli použiteľné! Preto si zavolali na pomoc známeho to poručíka *Columba*, ktorému postupne prezentujú **3 dôkazy** o vine jedného z podozrivých. Columbo uzná za vinného toho z podozrivých, o ktorého vine svedčí viac dôkazov.

Vlákná: Polícia
 Columbo

Iné: dôkaz náhodne nadobúda hodnotu 0 (nepoužiteľný dôkaz), 1 (pre podozrivého 1) alebo 2 (pre podozrivého 2) a ukladá sa do buffera (max. 3 dôkazy).

Synchronizácia: ak je buffer plný polícia nevydáva dôkazy, ak je buffer prázdny Columbo nemá čo skúmať a pýta si nové dôkazy.



Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 8 – Teletubbies

Dve individúá z gangu Teletubbies (konkrétne sa jedná o Tinky-Winky a Laa-Laa) sa hádajú. Hádajú sa o tom, či bude viac ako 60 % z 1000 náhodne vygenerovaných čísel nepárnych. Ak bude, vyhráva Tinky-Winky, ak nebude, vyhráva Laa-Laa.

Vytvorte pre Teletubbies C/C++ aplikáciu, ktorá potvrdí výhru jedného z nich. Teletubbies vlastnia mažiar na čísla, do ktorého sa zmestí 8 čísel naraz. Tinky-Winky do tohto mažiaru generuje 1000 náhodných čísel. Laa-Laa ich z tohto mažiaru vyberá a hodnotí, či ide o párne alebo nepárne číslo. Nezabudnite potvrdiť víťazstvo jedného z nich!

Bodovanie:

- návrh potrebných dátových štruktúr: 2 body;
- správna synchronizácia: 6 bodov;
- Tinky-Winky (producent): 6 bodov;
- Laa-Laa (konzument): 6 bodov;
- celková správnosť riešenia: 10 bodov (z toho 2 body za komentáre).

V prípade, že sa Vaše zadanie **nebude dať skompilovať** na stroji frios2.fri.uniza.sk, automaticky dostávate **0 bodov**.



Počas programovania môžete využiť nasledujúce dokumentácie:

- jazyk C:
 - <http://en.cppreference.com/w/c>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cprogramming/>
- jazyk C++:
 - <http://en.cppreference.com/w/cpp>
 - <http://www.cplusplus.com/reference/>
 - <https://www.tutorialspoint.com/cplusplus/>
- manuálové stránky Linuxu:
 - <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 9 – Pekáreň

[20b, min 12b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh pekárne [6b]**: Na začiatku sa pripraví **n** zákazníkov (ak sa n nezdaná parametrom, tak n=10), každý so svojim unikátnym identifikačným číslom a príslušnými dátami a pripraví sa Pekár so svojimi príslušnými dátami. Následne sa odštartuje beh pekárne výpisom informácie (napr. „Pekáreň otvorená“) a **čaká** sa na koniec pracovnej doby (keď sú všetci zákazníci obslužení) ktorý je opäť ohlásený výpisom informácie (napr. „Pekáreň zatvorená“). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Zákazník [2+5b]: Jeho úlohou je po otvorení pekárne sa dostať do pekárne, čo mu trvá náhodne od **2** do **6** sekúnd (rovnomerne rozdelené). Následne sa pokúsi odobrať si chlieb z pultu. Pokiaľ však žiaden chlieb nie je k dispozícii, tak zákazník čaká, kým nebude nejaký chlieb dodaný na pult. Ak je k dispozícii chlieb, tak si ho zákazník kúpi a jeho vykonávanie končí **[5b]**. Počas behu zákazníka je potrebné **vhodne vypisovať informácie o jeho činnosti**. Zákazník by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta **[2b]**.

Pekár [2+5b]: Jeho úlohou je po otvorení pekárne napiecť **n** chlebov, čo mu trvá **k** sekúnd (pokiaľ nie je k zadane, tak k=4). Následne vykladá na pult napečené chleby. Na pult sa zmestia **maximálne 2 chleby**. Pokiaľ už pekár nemá kde vykladať chlieb, tak čaká, kým sa mu uvoľní miesto na ďalší chlieb. Ak vyloží všetkých **n** chlebov, tak jeho činnosť končí **[5b]**. Počas práce pekára je potrebné **vhodne vypisovať informácie o jeho činnosti**. Pekár by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta **[2b]**.

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate za test 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 10 – Bar

[20b, min 12b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh baru [6b]**: Na začiatku sa pripraví **n** zákazníkov (ak sa n nezdaná parametrom, tak $n=10$), každý so svojim unikátnym identifikačným číslom a príslušnými dátami a pripraví sa Barman so svojimi príslušnými dátami. Následne sa odštartuje beh baru výpisom informácie (napr. „Bar otvorený“) a **čaká** sa na koniec pracovnej doby (keď sú všetci zákazníci obslužený), ktorý je opäť ohlásený výpisom informácie (napr. „Bar zatvorený“). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Zákazník [2+5b]: Jeho úlohou je po otvorení baru sa dostať do baru, čo mu trvá náhodne od **1** do **8** sekúnd (rovnomerne rozdelené). Následne sa pokúsi odobrať si drink z barového pultu. Pokiaľ však žiaden drink nie je k dispozícii, tak zákazník čaká, kým nebude nejaký drink dodaný na barový pult. Ak je k dispozícii drink, tak si ho zákazník kúpi a jeho vykonávanie končí **[5b]**. Počas behu zákazníka je potrebné **vhodne vypisovať informácie o jeho činnosti**. Zákazník by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta **[2b]**.

Barman [2+5b]: Jeho úlohou je po otvorení baru urobiť **n** drinkov. Postupne vykladá na pult hotové drinky, pričom vytvára **2 druhy** drinkov. Namiešanie prvého drinku si zvolí s pravdepodobnosťou **50%** a jeho príprava trvá **2** sekundy. Pokiaľ si nezvolí prípravu prvého drinku, tak automaticky pripravuje druhý drink, čo mu trvá **1** sekundu. Na barový pult sa zmestí **maximálne 1 drink**. Pokiaľ už barman nemá kde vykladať drink, tak čaká, kým sa mu uvoľní miesto na ďalší drink. Ak vyloží všetkých **n** drinkov, tak jeho činnosť končí **[5b]**. Počas práce barmana je potrebné **vhodne vypisovať informácie o jeho činnosti**. Barman by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta **[2b]**.

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 11 – Pátrač Otsok Shavk

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh úlohy v hre Pátrač Otsok Shavk (POS) [8b]**: Na začiatku sa pripraví n detektívov (ak sa n nezadá parametrom, tak $n=10$), každý so svojim unikátnym identifikačným číslom a príslušnými dátami a pripraví sa hľadaniu podozrivý Kefas Aldes so svojimi príslušnými dátami. Následne sa odštartuje hľadanie výpisom informácie (napr. „Nájdite Kefasa, pátranie sa začína!“) a čaká sa na koniec pátrania (keď všetci detektívi nájdu dôkaz), ktorý je opäť ohlásený výpisom informácie (napr. „Dôkazy boli zozbierané!“). Po skončení hľadania je potrebné vypísať informáciu, či sa podarilo nájsť Kefasa Aldesa (podarilo sa ho nájsť, pokiaľ sa našiel aspoň 1 platný dôkaz). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Detektív [3+10b]: Jeho úlohou je po začatí pátrania sa dostať k poslednému známemu miestu výskytu hľadanej osoby, čo mu trvá od 5 po 10 sekúnd (rovnomerne rozdelené). Po ceste na miesto má šancu na 40% stretnúť seržanta Arapata, ktorý mu dá tajnú vec menom Arch. Na mieste sa detektív pokúsi nájsť dôkaz. Pokiaľ však žiaden dôkaz nie je k dispozícii, tak detektív čaká, kým nebude nejaký dôkaz k dispozícii. Ak je k dispozícii dôkaz, tak si ho detektív zoberie a vráti sa späť na inšpektorát, čo mu trvá od 4 do 10 sekúnd (rovnomerne rozdelené). Po návrate a odovzdaní dôkazu sa jeho vykonávanie končí. Počas cesty späť je šanca stretnúť spojenca hľadanej osoby plukovníka Načbara (50%), ktorý zamení detektívovi platný dôkaz za neplatný, pokiaľ detektív má platný dôkaz a nevlastní Arch. [8b]. Počas vykonávania detektíva je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Detektív by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [3b].

Hľadaný podozrivý Kefas Aldes [2+7b]: Jeho úlohou je po spustení hľadania pripraviť n dôkazov (platný dôkaz na 10%, neplatný na 90%), čo mu trvá k sekúnd (pokiaľ nie je k zadané, tak $k=6$). Následne umiestňuje pripravené dôkazy po jednom na miesto, kam kráčajú detektívi. Na miesto sa zmesť maximálne 5 dôkazov. Pokiaľ už Kefas nemá kde dávať dôkazy, tak čaká, kým sa mu uvoľní miesto na ďalší dôkaz. Ak umiestni všetkých n dôkazov, tak jeho činnosť končí [5b]. Počas práce Kefasa je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Kefas by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 12 – Dovoz čaju pre Britské impérium

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh dodávok čaju pre Britské impérium [10b]**: Na začiatku sa pripraví n dodávateľov (ak sa n nezadá parametrom, tak $n=3$), každý so svojim unikátnym identifikačným číslom, čestnosťou (65% čestný dodávateľ, 35% nečestný dodávateľ), počtom dodávok (od 5 po 20, rovnomerne rozdelené) a príslušnými dátami a pripraví sa čajová komisia so svojimi príslušnými dátami. Následne sa odštartuje beh dodávok výpisom informácie (napr. „Príchod dodávok čaju započal! Splendid!“) a čaká sa na koniec dodávok (keď všetci dodávatelia vyčerpajú zásoby), ktorý je opäť ohlásený výpisom informácie (napr. „Koniec dodávok čaju! Je tu koniec sezóny.“). Po skončení sa zozbierajú informácie komisie a zobrazí sa vierohodnosť dodávateľov (číslo dodávateľa, rozhodnutie o čestnosti dodávateľa, percentuálna hodnota dobrých dodávok). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Dodávateľ [2+8b]: Jeho úlohou je po spustení požiadavky na dodávanie začať dodávať čaj do jednej dodávky do skladu. Čestný dodávateľ dodáva len kvalitný čaj. Pokiaľ sa jedná o nečestného dodávateľa, tak ten dodáva kvalitný čaj s pravdepodobnosťou 35% a nekvalitný s pravdepodobnosťou 65%. Presun do a zo skladu trvá 5 sekúnd pre dodávateľa s párnym číslom a 8 sekúnd pre dodávateľa s nepárnym číslom. Kapacita skladu je 15 dodávok. Pokiaľ je v sklade voľné miesto, dodávateľ tam vyloží svoj náklad, inak čaká, kým sa uvoľní miesto. Po odovzdaní všetkých dodávok dodávateľ končí [6b]. Počas behu dodávateľa je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Dodávateľ by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Čajová komisia [2+8b]: Jej úlohou je po začatí posielania dodávok ich preberať a kontrolovať. Komisia postupne vyberá zo skladu dodávky a preveruje ich, čo jej trvá 2 až 5 sekúnd (rovnomerne rozdelené). Pokiaľ sa v sklade nenachádza momentálne žiadna dodávka, tak komisia čaká. Výsledok preverenia si zapisuje a pokiaľ je to kvalitný čaj, tak ho posieľa Britskému impériu a pokiaľ to je nekvalitný čaj, tak ho predá inému odberateľovi. Po skontrolovaní všetkých očakávaných dodávok komisia končí [6b]. Počas práce komisie je potrebné **vhodne vypisovať informácie o jej činnosti [2b]**. Komisia by mala mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 13 – Počítač s SM Minmonsom

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh práce počítača s SM Minmonsom** [6b]: Na začiatku sa pripraví vlákno aktualizátora, vlákno používateľa a vlákno procesora, každé so svojimi príslušnými dátami a pamäť, ktorá má veľkosť k (k sa zadáva ako argument programu, ak sa nezadá, tak $k = 5$). Následne sa odštartuje fungovanie PC výpisom informácie (napr. „Konnichiwa! Ore wa SM Minmons desu!“) a čaká sa na koniec fungovania (keď sa spracujú všetky požiadavky), ktorý je opäť ohlásený výpisom informácie (napr. „Sayonara!“). Po skončení fungovania PC je potrebné vypísať informáciu, koľko % spracovaných požiadaviek predstavovali aktualizácie a koľko % predstavovali požiadavky používateľa. **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Aktualizátor [2+6b]: Jeho úlohou je po spustení PC začať posielat' n aktualizácií (čas medzi aktualizáciami 1-2 sekundy rovnomerne rozdelené, n je zadávané ako argument programu, ak sa nezadá, tak $n = 30$). Tieto aktualizácie prichádzajú do pamäte. Pokiaľ v pamäti nie je miesto alebo s ňou niekto pracuje, tak aktualizátor čaká, kým nebude nejaká časť pamäte k dispozícii. Po odoslaní všetkých aktualizácií sa jeho vykonávanie končí. [5b]. Počas vykonávania aktualizátora je potrebné **vhodne vypisovať informácie o jeho činnosti [1b]**. Aktualizátor by mal mať vhodné vstupné a výstupné dáta [2b].

Používateľ [2+6b]: Jeho úlohou je po spustení PC pracovať. Celkovo by mal odrobiť aspoň 30 sekúnd na PC, aby mohol skončiť. Preto postupne vykonáva úkony (trvanie úkonu 2-8 sekúnd rovnomerne rozdelené) až kým nedosiahne potrebný čas. Každý úkon je poslaný na spracovanie do pamäte. Preto ak tam nie je miesto alebo ju niekto iný používa, tak používateľ musí čakať (čakanie sa nezaráta do celkového času) [5b]. Počas práce používateľa je potrebné **vhodne vypisovať informácie o jeho činnosti [1b]**. Používateľ by mal mať taktiež vhodné vstupné a výstupné dáta [2b].

Procesor [2+6b]: Jeho úlohou je po spustení PC spracovať všetky požiadavky z pamäti. Pokiaľ je pamäť prázdna alebo ju používa niekto iný, tak procesor čaká. Pri spracovávaní požiadaviek si procesor zaznačí, aký typ požiadavky to bol. Spracovanie požiadavky mu trvá 1-3 sekundy rovnomerne rozdelené. Po spracovaní všetkých požiadaviek procesor končí [5b]. Počas práce procesora je potrebné **vhodne vypisovať informácie o jeho činnosti [1b]**. Procesor by mal mať taktiež vhodné vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 14 – Lovci morských tvorov

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh práce lovcov morských tvorov [9b]**: Na začiatku sa pripraví n lovcov (ak sa n nezadá parametrom, tak n=8), každý so svojim unikátnym identifikačným číslom a príslušnými dátami a pripraví sa hlavný kontrolór Rusu Naku so svojimi príslušnými dátami. Následne sa odštartuje lov výpisom informácie (napr. „Lov schválený, ide sa na to!“) a čaká sa na koniec lovu (keď všetci lovci ukončia lov), ktorý je opäť ohlásený výpisom informácie (napr. „Lov skončil! Dobrá práca! “). Po skončení lovu je potrebné vypísať štatistiku lovu (typ tvora, počet ulovených kusov, percentuálny podiel). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Lovec [2+9b]: Jeho úlohou je po začatí lovu odísť z centrálu na špecifické miesto (trvanie 2-4 sekundy, rovnomerné rozdelenie). Tam môže začať s lovom k morských tvorov (ak sa k nezadá, tak k=6). Celkovo je možné uloviť 5 rôznych tvorov s rôznou vzácnosťou (vyššie číslo, nižšia vzácnosť): 1 – tuniak (5%), 2 – langusta(10%), 3 – losos(15%), 4 – ustrica(20%), 5 – krab(50%). Následne sa lovec vracia s úlovkom na centrálu (nezabudnite na čas presunu), kde sa pokúsi svoj úlovok odovzdať. Pokiaľ už nie je miesto na odovzdanie alebo niekto iný je pri mieste odovzdania, tak lovec čaká. Inak sa odovzdá úlovok a lovec ide znova na lov. Práca lovca končí po ulovení všetkých k tvorov [7b]. Počas vykonávania lovca je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Lovec by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Hlavný kontrolór Rusu Naku [2+8b]: Jeho úlohou je po spustení lovu kontrolovať úlovky, zapísať si informáciu o danom úlovku a odložiť ich do príslušných prepraviek, čo mu trvá 4-6 sekúnd (rovnomerné rozdelenie). Pokiaľ kontrolór nenájde žiadny úlovok na centrále alebo je tam niekto iný, tak čaká, kým mu pribudne úlovok na centrále, ktorý potom odoberie na kontrolu. Ak skontroluje všetky úlovky, tak jeho činnosť končí [6b]. Počas práce kontrolóra je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Kontrolór Rusu Naku by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 15 – Zber čaju pre Britské impérium

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh zberu čaju pre Britské impérium [9b]**: Na začiatku sa pripraví n plantážnikov (ak sa n nezadá parametrom, tak n=10), každý so svojim unikátnym identifikačným číslom, pracovnou triedou (40% sedliak -> trieda 1, 35% sedlák -> trieda 2, 25% sedláček -> trieda 3) a príslušnými dátami, pripraví sa čajový kontrolór firmy Cevon A. J. Tea s.r.o., so svojimi príslušnými dátami a pripraví sa plantáž s kapacitou 60 dodávok čaju. Následne sa odštartuje beh zberu čaju výpisom informácie (napr. „Cevon [A]lways [J]oyful Tea - harvesting has begun! Splendid!“) a čaká sa na koniec zberu čaju (keď sa čajová plantáž celá spracuje), ktorý je opäť ohlásený výpisom informácie (napr. „End of harvesting! Tea time!“). Po skončení sa zozbierajú informácie kontrolóra a zobrazí sa úspešnosť zberu (počet odovzdaných dodávok, počet vhodných dodávok, počet nevhodných dodávok, percentuálna hodnota dobrých dodávok). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Plantážnik [2+9b]: Jeho úlohou je po spustení požiadavky na zber sa dostaviť z prístavu na plantáž (trieda 1 -> 4 sekundy, trieda 2 -> 6 sekúnd, trieda 3 -> 3 sekundy) a tam začať so zberom čajových lístkov do dodávky (trieda 1 -> trvanie 1000µs, trieda 2 -> trvanie 2000µs, trieda 3 -> 1500µs, čakanie v kritickej sekcii!). Musí platiť, že v jednom čase môže byť iba jeden plantážnik na plantáži. Po zbere nasleduje cesta do prístavu po inej ceste (trieda 1 -> 2 sekundy, trieda 2 -> 3 sekundy, trieda 3 -> 3 sekundy). Pokiaľ sa jedná o plantážnika sedláčka, tak ten pri ceste do prístavu vysype obsah dodávky a zamení ho za kávu. Dodávka je odovzdaná do skladu v prístave, ktorého kapacita je obmedzená na 5 dodávok. Pokiaľ je v sklade voľné miesto a nikto tam nie je, plantážnik tam vyloží svoj náklad, inak čaká, kým sa uvoľní miesto. Po vytážení plantáže plantážnik končí [7b]. Počas behu plantážnika je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Plantážnik by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Čajový kontrolór [2+8b]: Jeho úlohou je po začatí zberu kontrolovať dodávky dovezené plantážnikmi. Kontrolór postupne vyberá zo skladu dodávky a preveruje ich, čo mu trvá 2 sekundy pri vhodnej dodávke a 4 sekundy pri nevhodnej objednávke. Pokiaľ sa v sklade nenachádza momentálne žiadna dodávka alebo sa v sklade niekto aktuálne nachádza, tak kontrolór čaká. Výsledok preverenia si zapisuje a pokiaľ je to vhodná dodávka, tak ju nakladá na loď Britského impéria a pokiaľ to je nevhodná dodávka (káva), tak horko zaplače a zásielku posielá do Talianska. Po skontrolovaní všetkých očakávaných dodávok kontrolór končí [6b]. Počas práce kontrolóra je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Kontrolór by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 16 – Hamutex Malloku

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

S príchodom Vianoc sa manažér populárnej virtuálnej superstár Hamutex Malloku rozhodol rozbehnúť **šnúru koncertov s vianočnou tematikou**. Preto Vás poprosil o vytvorenie **programu za účelom simulácie tejto problematiky**.

Manažér postupne (náhodne v rozmedzí 2 až 4 hodín, rovnomerne rozdelené) prijíma náhodné požiadavky (rovnomerne rozdelené) na koncert od rôznych zadávateľov. Títo zadávatelia sú reprezentovaní číslami od 0 po 6, pričom toto číslo reprezentuje prioritu daného zadávateľa (čím vyššie číslo, tým vyššia priorita). Celkovo manažér prijme k požiadaviek (k musí byť väčšie ako 6 a menšie ako 18 a ak sa k zadá chybné alebo sa nezadá, tak $k = 10$). Potom manažér posieľa tieto požiadavky technickému personálu do špeciálnej schránky. Do tejto schránky sa zmestia najviac 3 požiadavky. Pokiaľ nemá manažér kde umiestniť požiadavku, tak čaká, kým sa mu uvoľní miesto pre požiadavku. Po odovzdaní všetkých požiadaviek manažér končí.

Technický personál má za úlohu dohliadať na priebeh koncertov. Keďže na koncert je potrebná špeciálna technika, je možné, aby sa v jednom čase konali maximálne 3 koncerty. Preto je potrebné zabezpečiť, aby sa odohralo všetkých k koncertov v čo najkratšom čase a pritom dbať na priority zadávateľov. Do simulácie je potrebné zahrnúť čas potrebný na dopravu a na koncert, ktorý je náhodný a rovnomerne rozdelený. Pre zadávateľov 0 až 2 je jeho trvanie od 1 do 5 hodín, pre zadávateľov 3 a 4 od 2 do 6 hodín a pre zadávateľov 5 a 6 od 4 do 8 hodín (pri generovaní berte do úvahy len hodiny). Po odohraní všetkých k koncertov má technický personál voľno. Pokiaľ sa v schránke nenachádza žiadna požiadavka, tak technický personál čaká na ďalšiu požiadavku.

Pre rýchlosť času použite mierku 1 hodina = 1 sekunda v programe. Na záver tiež vypíšte počty koncertov u jednotlivých dodávateľov s % podielom (napr.: zadávateľ 1 – 5 koncertov z 10, podiel 50%)

Bodovanie:

- Návrh potrebných dátových štruktúr: **3b**
- Správna inicializácia a uvoľnenie synchronizačných primitív: **2b**
- Správne implementovaný technický personál + výpisy: **8+2b**
- Správne implementovaný manažér + výpisy: **6+2b**
- Celková správnosť riešenia: **7b**

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 17 – BrandGames

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Firma **BrandGames** vyrába spoločenské hry. Spoločnosť zamestnáva n **pracovníkov** (argument programu, v prípade, že nie je zadáný, $n=10$), ktorí majú za úlohu pripravovať nápady na nové hry. Tieto následne spíšu a zanesú hlavnej manažérke Catanrine na stôl. Tá ich spracúva dávkovo, t.j. zoberie 5 pripravených materiálov a ohodnotí známku od 0 po 5. V prípade, že priemerné ohodnotenie nápadov v aktuálne spracúvanom balíku nápadov bude viac ako 3, tento balík sa použije na tvorbu novej hry. Inak sa celý balík nápadov zahodí. Na stôl sa zmestí maximálne k nápadov (argument programu, v prípade, že nie je zadáný, $k=10$). V prípade, že stôl je plný, pracovníci čakajú, kým sa neuvoľní. Vymýšľanie nápadov je veľmi namáhavá činnosť, ktorá zaberie 3 sekundy. Taktiež to dotyčného pracovníka natoľko vyčerpá, že po zanesení nápadu si musí dať ďalšie 2 sekundy prestávku. Jeden pracovník je schopný vyprodukovať maximálne m nápadov (argument programu, v prípade, že nie je zadáný, $m=5$). Po ich vyprodukovaní firmu opúšťa.

Manažérka Catanrina je veľmi vyťažená a kontrole nápadov sa nemôže venovať častejšie ako každých 5 sekúnd. Spracovanie jednej hromady nápadov jej trvá 2 sekundy. Počas tohto času ju nesmie nik rušiť a teda nemôže nikto prinášať nové nápady (!!sleep v kritickej sekcii!!). V prípade, že nemá na stole dostatočný počet nápadov, čaká a svoju nespokojnosť s lenivými zamestnancami, ktorí ju len okrádajú o čas, vyjadrí nedehtonstujúcim spôsobom do konzoly. Ohodnotenie nápadu prebieha veľmi kvalitným spôsobom. Manažérka si hodí dvomi kockami. Rozdiel týchto hodov určuje výslednú známku daného nápadu (je vhodné použiť funkciu `abs()` na určenie rozdielu hodov). Úloha manažérky končí, keď všetci zamestnanci odišli a ona spracovala všetky nápady, ktoré dostala. V prípade, že v hromade ostali nápady, ktorých je menej ako dávka na spracovanie, zahodí ich.

Na konci vypíšte, koľko nových spoločenských hier sa podarilo firme **BrandGames** vytvoriť.

Bodovanie:

- Návrh potrebných dátových štruktúr: **2b**
- Správna inicializácia a uvoľnenie synchronizačných primitív: **3b**
- Správne implementovaný zamestnanec: **8b**
- Správne implementovaná manažérka: **10b**
- Vhodné výpisy počas behu programu: **2b**
- Celková správnosť riešenia: **5b**

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

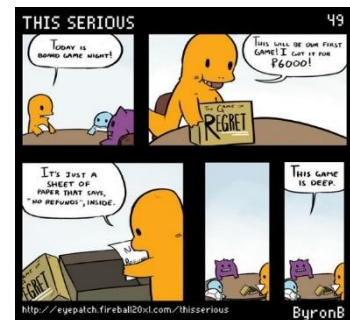
- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 18 – Plains of Shavk

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh úlohy v hre Plains of Shavk (POS) [8b]**: Na začiatku sa pripraví n hrdinov (ak sa n nezadá parametrom, tak $n=12$), každý so svojim unikátnym identifikačným číslom, zameraním (bojovník 50%, mág 20% a vrah 30%) a príslušnými dátami a pripraví sa tajomný liečiteľ Kán S'ur so svojimi príslušnými dátami. Následne sa odštartuje úloha výpisom informácie (napr. „Get the resurrection potion! Quest is now active!“) a čaká sa na koniec úlohy (keď všetci hrdinovia splnia úlohu) ktorý je opäť ohlásený výpisom informácie (napr. „Get the resurrection potion! Quest is now finished!“). Po skončení úlohy je potrebné vypísať štatistiku o počte získaných predmetov pre každé zameranie. **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Hrdina [3+10b]: Jeho úlohou je po začatí úlohy sa dostaviť k tajomnému liečiteľovi, čo mu trvá ako bojovníkovi od 1 do 4 sekúnd, ako mágovi 3 až 6 sekúnd a ako vrahovi 1 až 6 sekúnd (rovnomerne rozdelené). Následne sa hrdina pokúsi odkúpiť lektvar z pultu. Pokiaľ však žiaden lektvar nie je k dispozícii, tak hrdina čaká, kým nebude nejaký lektvar vyložený na pult. Ak je k dispozícii lektvar, tak si ho hrdina kúpi a vráti sa späť k zadávateľovi úlohy, čo mu trvá ako bojovníkovi od 1 do 4 sekúnd, ako mágovi 4 až 8 sekúnd a ako vrahovi 1 až 8 sekúnd (rovnomerne rozdelené). Po návrate sa jeho vykonávanie končí. Počas cesty späť je šanca stretnúť legendárneho hrdinu Shád'uda (mág – 30%, vrah 20%, bojovník – 10%), ktorý daruje hrdinovi talizman Otsok-eldes. [8b]. Počas behu hrdinu je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Hrdina by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [3b].

Tajomný liečiteľ Kán S'ur [2+7b]: Jeho úlohou je po spustení úlohy pripraviť n lektvarov, čo mu trvá k sekúnd (pokiaľ nie je k zadané, tak $k=5$). Následne vykladá pripravené lektvary po jednom na pult. Na pult sa zmestia maximálne 3 lektvary. Pokiaľ už Kán S'ur nemá kde vykladať lektvary, tak čaká, kým sa mu uvoľní miesto na ďalší lektvar. Ak vyloží všetkých n lektvarov, tak jeho činnosť končí [5b]. Počas práce Kán S'ura je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Kán S'ur by mal mať taktiež vhodne navrhnuté vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

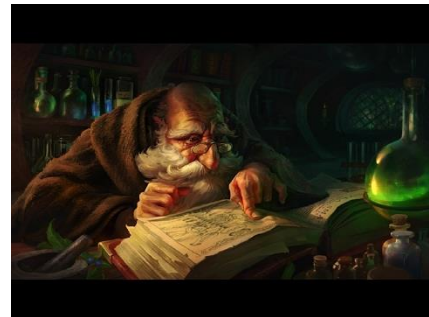
- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 19 – Voľby

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh volieb [10b]**: Na začiatku sa pripraví n voličov (ak sa n nezadá parametrom, tak $n=15$), každý so svojim unikátnym identifikačným číslom, volebnou preferenciou (65% Puttyn, 35% Ba\$h) a príslušnými dátami a pripraví sa volebná komisia so svojimi príslušnými dátami. Následne sa odštartuje beh volieb výpisom informácie (napr. „Voľby otvorené“) a čaká sa na koniec volieb (keď všetci voliči odvolia), ktorý je opäť ohlásený výpisom informácie (napr. „Voľby ukončené“). Po skončení volieb sa sčítajú hlasy a zobrazí sa výsledok volieb (meno, počet hlasov, percentuálna hodnota a výsledok). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Volič [3+8b]: Jeho úlohou je po otvorení volieb sa dostať do volebnej miestnosti, čo mu trvá náhodne od 1 do 10 sekúnd (rovnomerne rozdelené). Následne sa pokúsi odobrať si volebný hárok od volebnej komisie. Pokiaľ však žiaden hárok nie je k dispozícii, tak volič čaká, kým nebude nejaký hárok dodaný. Ak je k dispozícii volebný hárok, tak si ho volič zoberie, vyplní (od 100 do 150 μ s rovnomerne rozdelenie), odovzdá do volebnej urny a jeho vykonávanie končí. Je potrebné zabezpečiť, aby k urne mohol pristupovať len jeden volič v jednom okamihu **[6b]**. Počas behu voliča je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Volič by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta **[3b]**.

Volebná komisia [2+7b]: Jej úlohou je po otvorení volieb vyložiť n volebných hárkov. Komisia vykladá na stôl (na začiatku je stôl prázdny) volebné hárky až po maximálnu kapacitu stolu, ktorá je 5 hárkov. Pokiaľ už komisia nemá kde vykladať hárky, tak čaká, kým sa na stole nenachádzajú aspoň 2 volebné hárky. Ak komisia vyloží všetkých n hárkov, tak jej činnosť končí **[5b]**. Počas práce komisie je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Komisia by mala mať taktiež vhodné navrhnuté vstupné a výstupné dáta **[2b]**.

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

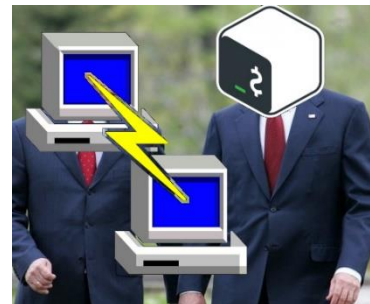
- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 20 – WaitChat

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

S príchodom Vianoc sa na čínskej sociálnej sieti WaitChat zvýšil počet chatujúcich používateľov. Keďže je potrebné, aby používatelia komunikovali medzi sebou správne a nepoužívali nevhodné slová či pejoratíva, bolo Vám predurčené, aby ste naprogramovali program simulujúci túto problematiku.

Kontrolór 八卦羞澀 má za úlohu kontrolovať prebiehajúcu komunikáciu medzi dvomi používateľmi 风笛 (Dudy) a 樱桃 (Cherry). Každá správa od používateľov je ukladaná do zásobníku správ, ktorý má obmedzenú kapacitu k (ak sa k nezadá ako argument programu, tak $k = 8$). Skôr než začne kontrolovať komunikáciu, musí sa na to psychicky a fyzicky pripraviť, čo mu trvá 5 sekúnd. Kontrolór potom berie 1 – 5 správ zo zásobníka správ a kontroluje ich, čo mu trvá 1 – 5 sekúnd (1 správa = 1 sekunda a zoberie minimálne 1 a maximálne 5 správ). Pokiaľ je zásobník prázdny, tak kontrolór čaká na príchod správy/správ. Ak kontrolór prečítal všetky správy, jeho vykonávanie končí výpisom informácie o znížení sociálneho kreditu oboch používateľov (meno, počet nevhodných správ, percentuálne vyjadrenie podielu nevhodných správ).

Používatelia Dudy a Cherry si spolu napíšu 50 správ (20 správ Dudy a 30 správ Cherry). Jednotlivé správy po napísaní putujú do zásobníka správ. Pokiaľ používatelia nemajú kde poslať správu (zásobník je plný), tak čakajú na uvoľnenie zásobníka. Napísanie jednej správy trvá náhodne od 1 do 3 sekúnd pre používateľa Dudy a od 1 do 2 sekúnd pre používateľa Cherry. Existuje určitá šanca (30 % Dudy a 50 % Cherry), že napísaná správa nebude vhodná. Po napísaní všetkých správ sa obaja používatelia odhlásia.

Pozor! Nedávajte sleep do kritickej sekcie, pokiaľ to nie je zadané. Všetky generované hodnoty sú rovnomerne rozdelené.

Bodovanie:

- Návrh potrebných dátových štruktúr: **2b**
- Správna inicializácia a uvoľnenie synchronizačných primitív: **3b**
- Správne implementovaný zamestnanec: **8b**
- Správne implementovaná manažérka: **10b**
- Vhodné výpisy počas behu programu: **2b**
- Celková správnosť riešenia: **5b**

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 21 – Predvianočný zhon

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Blížia sa Vianoce, čas príjemných chvíľ a zaslúženého pokoja. A pred týmto čarovným časom sa deje také šialenstvo, hnev a zúfalstvo, ktoré je nazývané ako nákup vianočných darčiekov. Pre pokoj svojich predajcov Vám riaditeľ odchodu Coond Joinota zadal vykonať simuláciu tohto zhonu.

Každých 10-15 minút (rovnorné rozdelenie) príde jeden človek (dobu strávenú v obchode zanedbáme). Každý človek kúpi 1 - 5 darčiekov (opäť rovnorné rozdelenie). V prípade, že v regáli nie je dostatok darčiekov pre jeho požiadavku, zoberie všetky darčeka čo tam sú a svoju nespokojnosť oznámi výpisom do konzoly (pri simulácii sa snažte obmedziť použitie hanlivých, teda pejoratívnych slov).

Kapacita regálu na darčeka je k (argument programu, minimálna hodnota 3). Obchod je otvorený 4 hodiny, potom už nemôže žiadny zákazník vstúpiť do predajne. Pracovníci obchodu nie sú zrovna moc motivovaní a preto sú schopní vyložiť len 2 darčeka za 5 minút a navyše začnú vykladať až keď počet vyložených darčiekov klesne pod $1/3$ kapacity (berte do úvahy len celočíselné delenie) a prestanú vykladať, keď počet darčiekov stúpne nad $2/3$. Na začiatku sú regály naplnené do $1/2$.

Pre rýchlosť času použijeme mierku 5 minút = 1 sekunda v programe.

Bodovanie:

- návrh potrebných dátových štruktúr: **3b**
- správna inicializácia a uvoľnenie synchronizačných primitív: **2b**
- správne implementovanie zákazníci: **10b**
- správne implementovanie pracovníci obchodu: **10b**
- celková správnosť riešenia: **5b**

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 22 – Káva pre doktoranda

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Istý doktorand na FRI veľmi obľubuje nápoj zvaný káva nielen kvôli jej účinkom na organizmus, ale najmä pre jej chuť. Každý deň prichádza niekoľkokrát do svojej obľúbenej kaviarne, kde si dopraje šálku tohto nápoja. Po nej sa usilovne venuje práci, čo dá najavo výpisom “zzzzzzz” do konzoly. Takto vydrží “pracovať” 2-5 hodín (rovnomé rozdelenie) a následne si opäť dopraje dúšok tohto nápoja.

V káve dominujú 2 základné chute – horkosť a kyslosť. V kaviarni zvyknú zmiešavať tieto 2 typy káv (horkú a kyslú) v náhodnom pomere. Káva sa mieša pri každej príprave zvlášť bez ohľadu na predchádzajúce pomery. V kaviarni nemajú príliš schopného baristu, a teda doba prípravy kávy zaberie 1 hodinu. Po tomto vyčerpávajúcom výkone si musí dať barista prestávku 3 hodiny.

Keďže tento doktorand preferuje horkejšie kávy, nevypije kávu, kde horkosť klesne pod 20 %, ale vyjadrí svoju nespokojnosť neurážlivým spôsobom.

Celková spokojnosť s kávou sa vypočíta ako priemerná hodnota spokojnosti s jednotlivými kávami a závisí výlučne od percentuálneho podielu horkej kávy (t. j. ak sa káva skladá z 80 % horkej a 20 % kyslej kávy, spokojnosť je 80 %). Ak klesne celková spokojnosť pod 40%, opäť vyjadrí svoju nespokojnosť a prestane navštevovať kaviareň úplne (kaviareň zbankrotuje a príklad končí).

Pre rýchlosť času použijeme mierku 1 hodina = 1 sekunda v programe.

Bodovanie:

- návrh potrebných dátových štruktúr: **3b**
- správna inicializácia a uvoľnenie synchronizačných primitív: **2b**
- správne implementovaný doktorand: **10b**
- správne implementovaný barista: **10b**
- celková správnosť riešenia: **5b**

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 23 – Výroba kávy

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Kým sa káva dostane do kaviarne, musí prejsť niekoľkými fázami. Plantážnici ju najskôr musia vypestovať a pozbierať. Následne sa káva prepraví do pražiareň, kde sa upraží a až potom sa predáva do kaviarní.

Plantážnici, od ktorých odoberá kávu naša pražiareň, sú schopní dodať 1 kg kávy každých 1 – 3 hodín. Pražiareň má kapacitu skladu 10 kg. V prípade, že sa sklad naplní, plantážnici idú na zaslúžený oddych a nevrátia sa do práce skôr ako o 6 hodín.

Pražiareň spracúva kávu dávkovo, teda počká, kým bude mať na sklade aspoň 5 kg kávy. Následne zoberie práve 5 kg a opraží ju. Chuť kávy, a teda aj jej cena, v značnej miere závisí práve od praženia. V našej pražiarni ju pražia náhodne od 1 do 5 hodín. V prípade, že sa káva bude pražiť krátko (do 2 hodín), cena za 1 kg bude 15 EUR. V prípade, že sa káva bude pražiť od 2 do 3 hodín, cena za 1 kg bude 20 EUR. V prípade, že doba praženia bude od 3 do 4 hodín, cena za 1 kg bude 30 EUR. Ak sa však káva bude pražiť dlhšie ako 4 hodiny, káva bude spálená a nebude možné ju predať za viac ako 5 EUR za 1 kg. Po upražení kávy si dajú všetci zamestnanci pražiareň oddych na 2 hodiny a po tomto čase sa môžu opäť vrátiť k praženiu.

Z plantáže je možné získať 500 kg kávy. Po vyčerpaní tohto množstva sa vypíše priemerný zisk pražiareň a program končí.

Pre rýchlosť času použijeme mierku 1 hodina = 1 sekunda v programe.

Bodovanie:

- návrh potrebných dátových štruktúr: **3b**
- správna inicializácia a uvoľnenie synchronizačných primitív: **2b**
- správne implementovanie plantážnici: **10b**
- správne implementovaná pražiareň: **10b**
- celková správnosť riešenia: **5b**

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 24 – Elektronické noviny

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Nemenovaná environmentálna aktivistka sa rozhodla privyrábať si na svoje osvetové aktivity predajom novín v miestnom novinovom stánku.

Na pult novinového stánku sa zmestí práve n novín (argument programu). Zákazníci prichádzajúci k pultu si kúpia 1 výtlačok a následne sa rozhodnú, či ho chcú v tlačenej alebo elektronickej podobe. Pravdepodobnosť, že sa rozhodnú pre elektronicкую podobu je 35 %. V prípade, že sa rozhodli pre tlačенú podobu, aktivistka, sklamaná ich prístupom k životnému prostrediu, ich jemne napomenie slovami "How dare you!!!". V prípade, že sa rozhodli pre elektronicкую podobu, odovzdajú výtlačok na zbernom mieste pri stánku a odnášajú si len kód, ktorý im umožní stiahnuť si noviny v elektronickej podobe (pozn. odovzdávanie týchto výtlačkov a následné stiahnutie si elektronických verzií nie je nutné implementovať v rámci tohto zadania). Ak zákazník príde k stánku a zistí, že tam nie je vyložený ani 1 výtlačok, sklamaný odchádza. Toto sklamanie vyjadri výpisom na konzolu.

Zákazníci prichádzajú k stánku každých 5 – 15 minút (rovnorné rozdelenie). Keďže aktivistka šetrí svoje sily, začne vykladať noviny až keď klesne ich vyložený počet pod $1/3$ kapacity pultu. Noviny sa vykladajú po 1 kuse rýchlosťou 1 ks / 5 minút. V prípade, že sa pult naplní doplna, aktivistka si spraví prestávku a začne vykladať opäť až keď klesne ich vyložený počet pod $1/3$ kapacity pultu.

Noviny v stánku sú v náklade k kusov (argument programu). Po ich vypredaní sa stánok zatvára a program končí. Určte, aké percento predaných novín bolo v elektronickej podobe.

Pre rýchlosť času použijeme mierku 5 minút = 1 sekunda v programe.

Bodovanie:

- návrh potrebných dátových štruktúr: **3b**
- správna inicializácia a uvoľnenie synchronizačných primitív: **2b**
- správne implementovaná aktivistka: **10b**
- správne implementovaní zákazníci: **10b**
- celková správnosť riešenia: **5b**

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 25 – Sušenie húb

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh práce sušičky húb [10b]**: Na začiatku sa pripraví n hubárov (ak sa n nezadá parametrom, tak $n = 5$), každý so svojim unikátnym identifikačným číslom a príslušnými dátami a pripraví sa hlavný sušič Mada Šád so svojimi príslušnými dátami. Následne sa odštartuje sušenie výpisom informácie (napr. „Čas sušenia nastal, hor sa na to!“) a čaká sa na koniec sušenia (keď všetky jedlé huby budú usušené), ktorý je opäť ohlásený výpisom informácie (napr. „Je nasušené! Paráda!“). Po skončení sušenia je potrebné vypísať štatistiku zárobkov hubárov (id hubára, zarobená suma v € ($100\text{ c} = 1\text{ €}$), percentuálny podiel z celkovej zarobenej sumy všetkých hubárov). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Hubár [2+8b]: Jeho úlohou je po začatí sušenia sa dostať zo svojho auta s jednou hubou k sušičovi, čo mu trvá od 1-2 s (rovnomerne rozdelené a tento čas sa bude používať vždy pri ceste k autu a od auta). Celkovo má každý hubár v aute 15 húb, pričom platí, že má buď jedlú hubu bedľu, šanca 25 %; jedlú hubu dubák, šanca 10 %; jedlú hubu kozák, šanca 25 % alebo jedovatú hubu muchotrávka, šanca 40 %. U sušiča sa pokúsi predať svoju hubu, pričom ju vyloží na pult. Pult má kapacitu na 5 húb. Pokiaľ už nie je miesto na odovzdanie, tak hubár čaká. Inak na pult položí svoju hubu, dostane príslušnú odmenu (dostane ju hneď po odložení huby na pult, pričom za dubák dostane 20 c, za bedľu dostane 10 c, za kozák dostane 5 c a za muchotrávku nedostane nič) a ide znova k autu pre ďalšiu hubu. Práca hubára končí po predaní všetkých svojich húb. **[6b]**. Počas vykonávania činnosti hubára je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Hubár by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta **[2b]**.

Hlavný sušič Mada Šád [2+8b]: Jeho úlohou je po spustení sušenia sušiť huby, zapísať si informáciu o ich type a odložiť ich do príslušných prepraviek, čo mu trvá 2 s na jednu jedlú hubu a 0 s pre jedovatú. Huby berie sušič z pultu. Pokiaľ sušič nenájde žiadnu hubu na pulte, tak čaká, kým mu pribudne huba na pulte. Sušič berie na sušenie toľko húb, koľko je na pulte. Po spracovaní všetkých húb jeho činnosť končí výpisom informácie o počte nasušených jedlých húb daných druhov **[6b]**. Počas práce sušiča je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Tiež by mal mať sušič vhodné navrhnuté vstupné a výstupné dáta **[2b]**.

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate za test 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

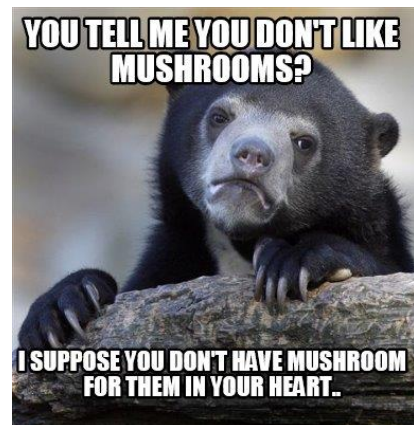
- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 26 – Sťahovacia spoločnosť

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh práce sťahovacej spoločnosti „There and not back again“ [6b]**: Na začiatku sa pripraví vlákno baliča, vlákno prepravcu a vlákno vykladača, každé so svojimi príslušnými dátami a 2 miesta na odkladanie balíkov, ktoré majú veľkosť k (k sa zadáva ako argument programu, ak sa nezadá, tak $k = 5$). Následne sa odštartuje sťahovanie výpisom informácie (napr. „So it begins“) a čaká sa na koniec fungovania (keď sa presunú všetky balíky), ktorý je opäť ohlásený výpisom informácie (napr. „It's done!“). Po skončení sťahovania je potrebné vypísať informáciu, koľko času bolo potrebné na jednotlivé časti (bez rátania času čakania). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Balič Balamir [2+6b]: Jeho úlohou je po odštartovaní začať baliť n balíkov (čas balenia 2-4 sekundy rovnomerne rozdelené, n je zadávané ako argument programu, ak sa nezadá, tak $n = 25$). Existuje šanca (15%) že čas balenia je dvojnásobný, keďže sa jedná o krehký náklad. Tieto balíky prichádzajú po jednom na 1. odkladacie miesto. Pokiaľ na odkladacom mieste nie je voľné miesto alebo tam pracuje prepravca, tak balič čaká, kým nebude nejaké miesto k dispozícii. Po pripravení všetkých balíkov sa jeho vykonávanie končí. [5b]. Počas vykonávania baliča je potrebné **vhodne vypisovať informácie o jeho činnosti [1b]**. Balič by mal mať vhodné navrhnuté vstupné a výstupné dáta [2b].

Prepravca Godalf [2+6b]: Jeho úlohou je po odštartovaní začať prepravovať balíky z 1. miesta do 2. miesta. Na prepravu môže použiť 2 dopravné prostriedky (Shadowfax – trvanie cesty tam 2-4 sekundy na balík, späť 1-2 sekundy rovnomerne rozdelené, šanca 75%; TakumiEx – trvanie cesty tam 2 sekundy a aj späť, šanca 25%), každé s maximálnou kapacitou 2 balíkov. Ak je 2. miesto zaplnené, tak prepravca čaká na uvoľnenie a ak je 1. miesto prázdne tak prepravca čaká na vloženie. Prepravca tiež nemôže ísť na miesto, keď ho iný člen používa [5b]. Počas práce prepravcu je potrebné **vhodne vypisovať informácie o jeho činnosti [1b]**. Prepravca by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta [2b].

Vykladač Freedo [2+6b]: Jeho úlohou je po odštartovaní doniesť balíky po jednom do nového domu. Pokiaľ je 2. miesto prázdne alebo ho používa prepravca, tak vykladač čaká. Vykladanie balíku mu trvá 1-3 sekundy rovnomerne rozdelené, pričom existuje 50% šanca že počas vykladania balíku strávi 4 sekundy navyše cestou po schodoch na poschodie. Po spracovaní všetkých požiadaviek vykladač končí [5b]. Počas práce vykladača je potrebné **vhodne vypisovať informácie o jeho činnosti [1b]**. Vykladač by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate za test 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 27 – Predaj hry

[30b, min 18b] Úlohou je vyriešiť nasledovný problém pomocou synchronizácie vlákien v jazyku C/C++:

Je potrebné simulovať **beh predaja a hodnotenia hry [10b]**: Na začiatku sa pripraví n hráčov (ak sa n nezadá parametrom, tak n=15), každý s príslušnými dátami a pripraví sa predajca firmy Stonerats so svojimi príslušnými dátami. Následne sa odštartuje beh predaja výpisom informácie (napr. „Predaj spustený, kúpte si svoje gca++!“) a čaká sa na koniec predaja (keď všetci hráči nakúpia a ohodnotia hru), ktorý je opäť ohlásený výpisom informácie (napr. „Predaj skončený! No refunds!“). Po skončení predaja sa spracujú hodnotenia hráčov a zobrazí sa skóre hry (priemerná hodnota, maximálne skóre a minimálne skóre). **Dbajte na to, aby ste inicializovali a na záver aj zničili všetky potrebné synchronizačné nástroje.**

Hráč [3+8b]: Jeho úlohou je po otvorení predaja sa dostať do predajne, čo mu trvá náhodne od 1 do 5 sekúnd (rovnomerne rozdelené). Následne sa pokúsi zakúpiť si hru. Hráč môže byť buď normálny hráč (65% zaplatí za gca++ verziu 60€) alebo zberateľ (35%, zaplatí za gca+++ verziu 75€). Pokiaľ však žiadna hra nie je k dispozícii, tak hráč čaká, kým nebude nejaká hra k dispozícii. Ak je k dispozícii hra, tak si ju hráč kúpi, vráti sa domov (rovnaký čas ako pri ceste do predajne), zahrá si ju (od 1 do 3 s rovnomerne rozdelenie) a následne vloží svoje hodnotenie (od 1 do 10 rovnomerne rozdelenie) do databázy recenzii a jeho vykonávanie končí. Je potrebné zabezpečiť, aby k databáze mohol pristupovať len jeden hráč v jednom okamihu [6b]. Počas behu hráča je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Hráč by mal mať taktiež vhodné navrhnuté vstupné a výstupné dáta [3b].

Predajca firmy Stonerats [2+7b]: Jeho úlohou je po otvorení predaja vyložiť n hier gca++/gca+++ (typ hry sa určí až pri zakúpení zákazníkom). Predajca vykladá na pult (na začiatku je pult prázdny) hry až po maximálnu kapacitu pultu, ktorá je 6. Pokiaľ už predajca nemá kde vykladať hry, tak čaká, kým sa na pulte neklesne počet hier pod 3 kusy. Ak predajca vyloží všetkých n hier, tak jeho činnosť končí a na záver vypíše celkovú vyzbieranú sumu [5b]. Počas práce predajcu je potrebné **vhodne vypisovať informácie o jeho činnosti [2b]**. Predajca by mala mať taktiež vhodné navrhnuté vstupné a výstupné dáta [2b].

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 28 – 2Drinkstone

V dedinke s príhodným názvom 2Drinkstone, keďže sa nachádza iba v 2 dimenziách, žijú okrem ostatných obyvateľov aj baníci a prospektori, ktorých činnosť sa dá popísať nasledovne:

Baník: z nástenky zoberie súradnice miesta, kde pôjde kopat' a následne sa vyberie na dané súradnice rýchlosťou 5dm (drinkstone meters)/s (t. j., ak je ruda na súradniciach [3;4], vzdialenosť bude $\sqrt{(3^2+4^2)}dm = 5dm$, takže cesta mu bude trvať 1s. Doba cestovania sa zaokrúhľuje na mikrosekundy.). Na nich sa nachádza náhodný počet kusov jednej z rúd (s pravdepodobnosťami ako sú v tabuľke), tieto vytáži a opäť sa vráti k nástenke rovnakou rýchlosťou. Počas spiatočnej cesty však môže baníka prepadnúť creepster (pravdepodobnosť stretu je 25%) – v takejto situácii baník stratí rudu naťaženú v tejto ťažbe. V prípade, že baník príde k nástenke a nenájde tam žiadne súradnice s miestom na ťaženie, čaká, kým tam prospektor nejaké vyvesí. Baník vykoná práve 20 ťažieb a potom skončí svoju činnosť. Na konci práce vypíše svoj celkový zárobok.

Prospektor: Notch Nossrep, hlavný prospektor tejto dediny generuje súradnice s možnou pozíciou rúd a následne tieto súradnice zavesí na nástenku, ku ktorej majú všetci baníci prístup. Prospektor je schopný vygenerovať 1 súradnice každé 1-3 sekundy. Keďže nemôže generovať mimo územia obce, všetky vygenerované súradnice musia byť max 20 v smere x <-20, 20> a 50 v smere y <-50, 50> (pozn. prípad, že prospektor vygeneruje viackrát tie isté súradnice budeme ignorovať – takže na jednom mieste sa môžu nachádzať rudy aj viackrát). V prípade, že je nástenka plná, prospektor negeneruje ďalšie súradnice, ale ide do miestneho lokálu, kde čaká, až kým sa nástenka úplne nevyprázdni. Prospektor končí svoju činnosť v prípade, že vygeneroval dostatočné množstvo súradníc – t. j. počet baníkov * 20.

Veľkosť nástenky a počet baníkov sú parametre programu.

Ruda	Pravdepodobnosť výskytu	Počet kusov na jednom mieste	Cena za kus
Železo	45%	<1;10>	5
Zlato	30%	<1;6>	15
Diamanty	15%	<1;3>	50
Nič	10%	-	-

Everyone: you need 8 hours of sleep every day bruh

Me playing minecraft at 3 am:



Hodnotenie: max 30b, min. 18b:

- návrh potrebných dátových štruktúr: 3 body;
- správne vytvorenie a uvoľnenie synchronizačných primitív: 2 body;
- producent: 10 bodov;
- konzument: 10 bodov;
- celková správnosť riešenia: 5 body.

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate za test 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>

Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 29 – Plavba na Oravskej priehrade

Cestovná kancelária ponúka plavby luxusnou loďou po Oravskej priehrade. Cieľom kancelárie je realizovať plavby, ktoré sú iba plne kapacitne obsadené. Kapacita lode a cena za lístok sa zadáva parametricky. Pasažieri prichádzajú v náhodných časových intervaloch, približne každých 5 sekúnd. No prichádzajú v skupinkách (rodina, kamaráti spolu). Priemerná veľkosť skupiny je 3. Pokiaľ je pasažier mladší ako 12 rokov, tak platí len 50% ceny lístka. Pravdepodobnosť, že pasažier je mladší ako 12 rokov je 35%. Po naplnení kapacity odchádza loď na plavbu.

Po naložení pasažierov sa loď vydá na plavbu. Plavba trvá približne 20 až 50 sekúnd, alebo pokiaľ sa loď nepotopí. Vzhľadom na zlý technický stav je pravdepodobnosť potopenia lode až 15 percent. Cestujúci počas plavby nakupujú jedlo. Cena jedla je v rozsahu 3 až 10 euro. Počas plavby si ukladajte sumu, ktorú cestujúci zaplatili za jedlo a počet predaných porcií. Po úspešnom dokončení plavby je loď pristavená do prístavu, pasažieri vystúpia a môžu nastúpiť noví pasažieri. Vystúpenie pasažiera trvá od 1 do 2 sekúnd. Ak sa loď potopí, tak aplikácia vypíše celkový dosiahnutý zisk, zisk za jedlo, a priemernú cenu predanej porcie.

Vytvorte aplikáciu v C/C++, ktorá bude riešiť tento problém formou producenta konzumenta. Pre predstavenie problému producenta konzumenta je možné využiť materiál dostupný na odkaze obrázku.

Hodnotenie: max 30b, min. 18b:

- návrh potrebných dátových štruktúr: 3 body;
- správne vytvorenie a uvoľnenie synchronizačných primitív: 3 body;
- producent: 10 bodov;
- konzument: 10 bodov;
- celková správnosť riešenia: 4 body.

V prípade, že sa Vaše zadanie nebude dať skompilovať, automaticky dostávate 0 bodov.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.

Zadanie 30 – Šialené odmeny

Mimozemská internetová spoločnosť ponúka online predaj cez svoj web a poskytuje dovoz objednávok do elektronických boxov na výdaj tovaru. Ponuka tovaru je prakticky neobmedzená, ale obmedzený je **výdajný box** svojou **kapacitou** podľa typu miesta **od 10 – 20 kusov** tovarov, ktoré sa v ňom môžu nachádzať. Ďalej ako benefit poskytuje za každý nákup **možnosť získať zlatky (5 % z ceny objednávky, ak je vyššia ako 12 Eur)**, ktoré sa dajú využiť ako zľava na ďalšie nákupy. Avšak obmedzený je **príjem** zberateľa zlatiek, ktorý je generovaný v rozsahu **od 1 800 do 7 500 Eur**. Tovar, ktorý objednáva, si kupuje v hodnote **od 4,20 do 42,00 EUR**. Objednávky vytvára priebežne dokiaľ nevyčerpá svoj príjem. Zaplatená objednávka je doručovaná ihneď do boxu, ak nie je plný. Ak sa box naplní, doručovanie je pozastavené, pokiaľ sa box úplne nevyprázdni. **Vyzdvihnutie objednávky** z boxu rieši zberateľ, pričom **nie vždy je možné** objednávku prevziať (**75 % prípadov**). Počas slnečného dňa je klávesnica na displeji skrytá a teda zberateľ **musí počkať** na lepšie environmentálne podmienky (chvilku počká) alebo **musí zavolať magické slovíčko „Mellon“**, aby sa klávesnica zobrazila. Na kúzelné slovíčko si spomenie v **45 % prípadov**.

Zberateľovo šialenstvo končí vyčerpaním prostriedkov alebo stavom, keď si zberateľ nemôže zo zostatku prostriedkov kúpiť vybraný tovar. Pri ukončení sa **vypíše bilancia zberateľa**, t. j., koľko tovarov objednal, koľko zlatiek nazbieral a koľkokrát si spomenul na magické slovíčko. Všetky náhodné veličiny sú generované z rovnomerného rozdelenia pravdepodobnosti.

Vytvorte aplikáciu v C/C++, ktorá bude riešiť tento problém producent-konzument.

Hodnotenie: max 30b, min. 18b

- návrh potrebných dátových štruktúr: 5 bodov;
- správne vytvorenie a uvoľnenie synchronizačných primitív: 3 body;
- producent: 10 bodov;
- konzument: 8 bodov;
- celková správnosť riešenia: 4 body.

Pozor! Je potrebné, aby bolo možné skompilovať Vami napísaný program! Pokiaľ to nebude možné, automaticky získavate 0b.

Počas programovania môžete využiť nasledujúce dokumentácie:

jazyk C:

- o <http://en.cppreference.com/w/c>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cprogramming/>

jazyk C++:

- o <http://en.cppreference.com/w/cpp>
- o <http://www.cplusplus.com/reference/>
- o <https://www.tutorialspoint.com/cplusplus/>

manuálové stránky Linuxu:

- o <https://linux.die.net/>



Pracujte každý samostatne! Akýkoľvek pokus o spoluprácu s inou osobou vedie automaticky k získaniu známky Fx z predmetu bez možnosti jej opravy.